Using ACS Data to Construct and Analyze Couple-Level Characteristics in Stata

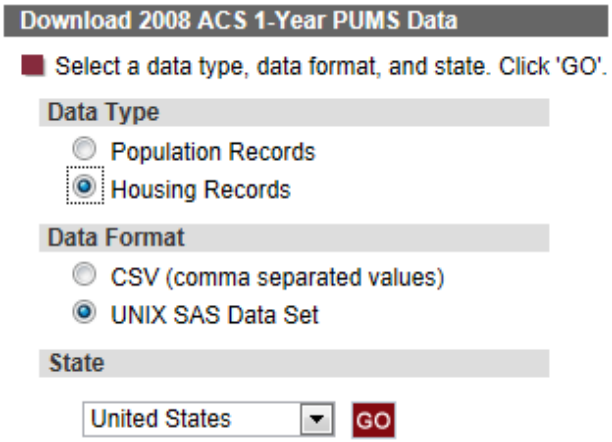
The American Community Survey (ACS) is a national survey that provides yearly data on households and individuals within households in the United States, including information on couples in coresidential (married or cohabiting) relationships. To determine whether individuals reside in same- and or opposite-sex coresidential relationships, it is necessary to identify individual's relationship to the household head, as well as the sex of both the individual and the household head. This information (relationship and sex) is contained on two different lines of data, complicating variable construction. This guide provides step-by-step instructions on how to manipulate the data to generate accurate indicators of coresidential relationships. Though providing only one example, the procedures outlined can be applied to other ACS variables.

The first sections of this guide outline the steps needed to obtain and prepare ACS data for analysis. Following these procedures, it will then provide detailed instruction on how to accurately construct indicators of same- and opposite-sex coresidential relationships. Please note that the ACS only records individual relationships with respect to the household head. Thus, you will not be able to use the ACS to identify the relationships between all individuals within a given household. The final section outlines the steps needed to generate weighted estimates of ACS data.

1. Obtain data from the ACS website:

   http://www.census.gov/acs/www/data_documentation/pums_data/

   a. Select which year(s) of data you want to use
      i. This example uses 2008 ACS 1-year PUMS
   b. Choose data type, data format, and state
      i. This example uses both the Housing and Population Records in SAS format for the entire United States
         - Note that you download the data in SAS format, though you will eventually be using Stata
      ii. Select the Housing Record first



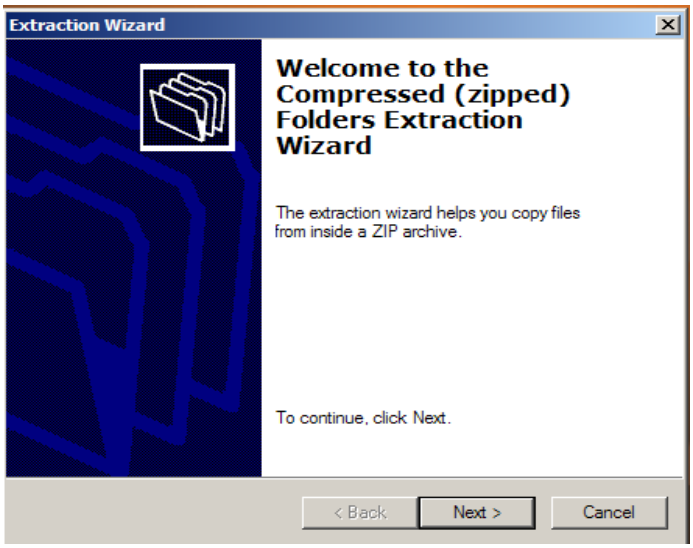
c. Click “Go”

d. Save to Desktop as pc_hus.zip (pc_pus.zip for Population Record)

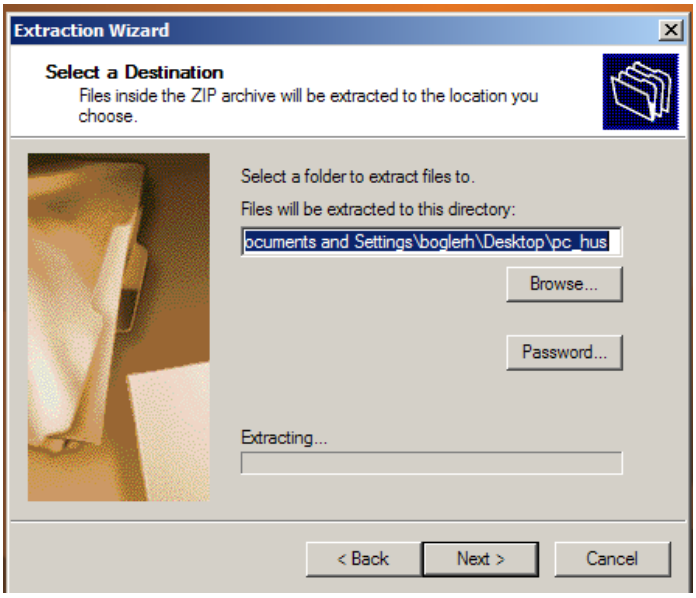
e. Right click the pc_hus.zip icon on the desktop

f. Click “Extract All…”

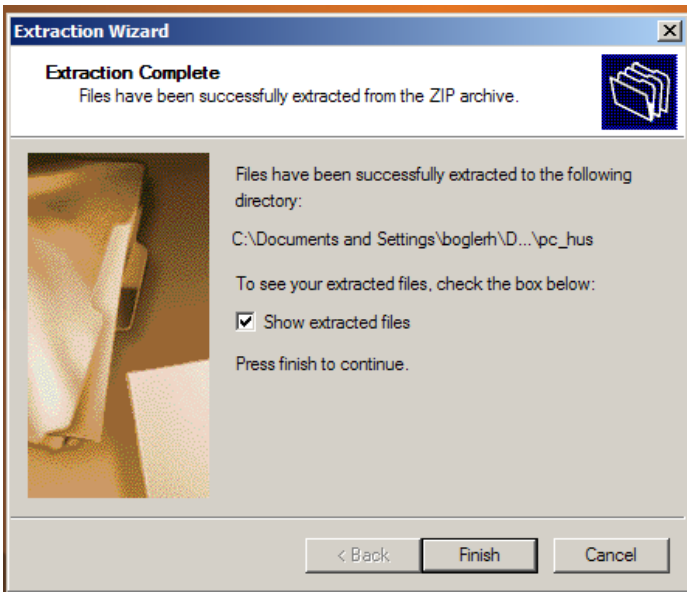
g. Follow the Extraction Wizard: Select “Next”



h. Assign location for file extraction, then click “Next”



i. Check “Show extracted files” box, then click “Finish”



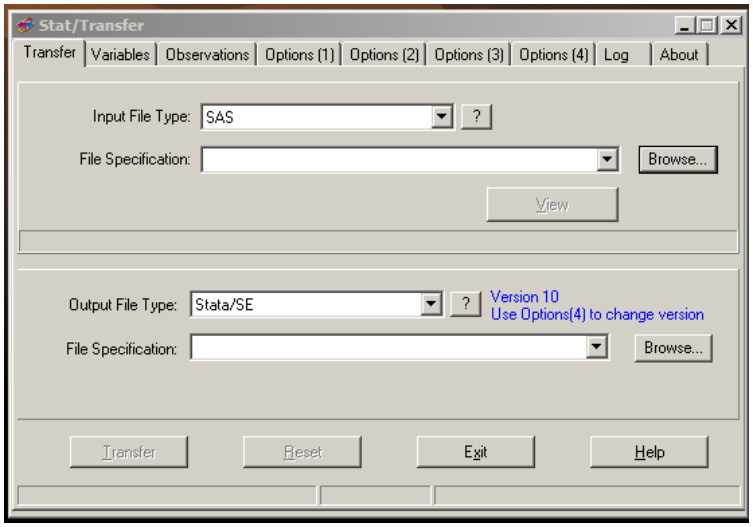
j. Once open, the folder should contain 4 files:

- 01ACS2008_PUMS_README.pdf
- ACS_2006_2008_PUMS_Re-release.pdf
- **psam_husa.sas7bdat** (**psam_pusa.sas7bdat** if Population Record)
- **psam_husb.sas7bdat** (**psam_pusb.sas7bdat** if Population Record)

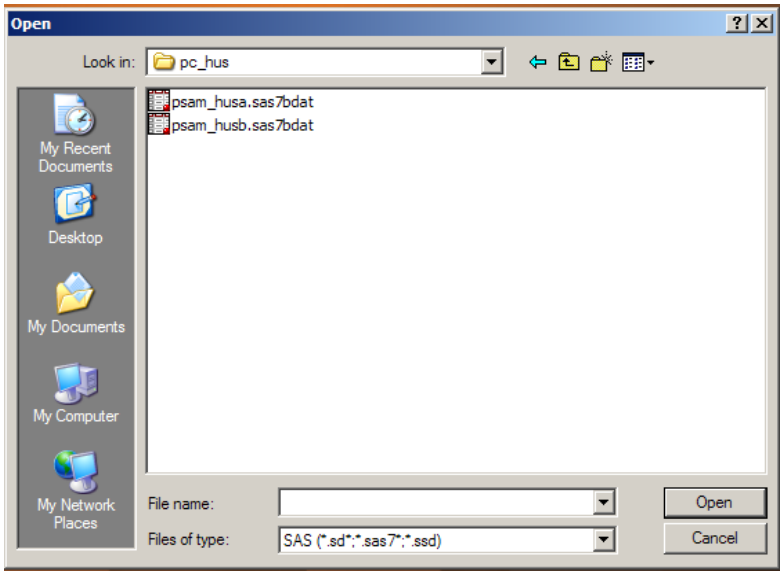
k. Repeat steps c through j using the Population Record

   i. Note that there are 2 Housing Records and 2 Population Records; this is due to the large size of the ACS

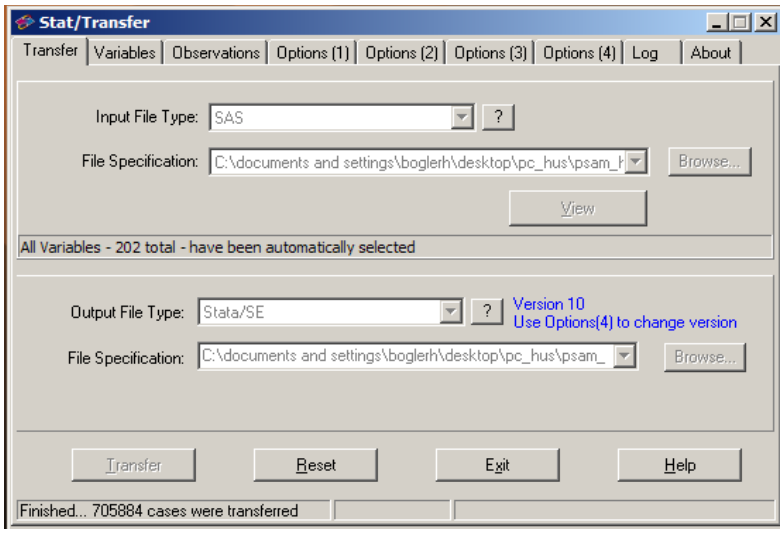
2. Use Stat/Transfer to convert your 4 data files (e.g., **psam_husa.sas7bdat**, **psam_husb.sas7bdat**, **psam_pusa.sas7bdat**, and **psam_pusb.sas7bdat**) into Stata format:
   a. Open Stat/Transfer
   b. Select SAS as "Input File Type" and Stata v. 10 as "Output File Type"
      i. The output file type will vary depending on which version of Stata you are using



   c. Click "Browse" and select psam_husa.sas



d. Click “Transfer”, and exit Stat/Transfer when the operation is completed



e. Repeat steps a through d for the remaining 3 files

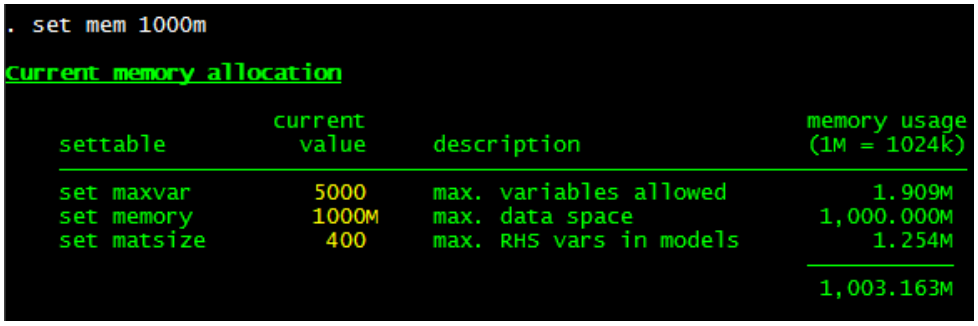
f. By the end of the operation you should have the following 4 files:

- **psam_husa.dta**
- **psam_husb.dta**
- **psam_pusa.dta**
- **psam_pusb.dta**

g. If you do not have access to Stat/Transfer please refer to the following, http://www.bgsu.edu/downloads/cas/file97319.pdf. This document will guide you on how to read CSV (comma separated values) data into Stata, the other data format provided by the ACS website.

3. Retain Key Variables:
   a. Due to the large size of the ACS you will want to give yourself a large memory (RAM) allocation. 1000-1200m is recommended, so you will want to close all non-essential programs.
      i. Stata code:

```
set more off
set mem 1000m
set maxvar 32767
```



```
. set mem 1000m

current memory allocation

                    current                                 memory usage
   settable          value      description                 (1M = 1024k)
   ------------------------------------------------------------------
   set maxvar         5000      max. variables allowed          1.909M
   set memory         1000M     max. data space             1,000.000M
   set matsize         400      max. RHS vars in models         1.254M
                                                            ----------
                                                            1,003.163M
```

   b. Given the large size of the data you will want to drop all non-essential variables. The essential variables for this example are outlined; however, the variables you select may change with your focus.
      i. For a complete list and description of variables found in both the Housing and Population Records, download the ACS data dictionary at http://www.census.gov/acs/www/data_documentation/pums_documentation/
   c. The Housing Record
      i. Essential variables:
         - Housing unit (`serialno`)
         - Number of person records following housing record (`np`)
         - Unmarried partner household (`partner`)
      ii. Stata code:

```
use "C:\psam_husa.dta"
keep serialno np partner
count
sort serialno
save "C:\psam_husa_001.dta"
```

```
. use "R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_hus\psam_husa.dta"

. keep serialno np partner

. count
705884

. sort serialno

. save "R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_hus\psam_husa_001.dta"
file R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_hus\psam_husa_001.dta saved
```

- Take note of results for the `count` command
- Perform again using "C:\psam_husb.dta"; results for `count` command should be equal to 667,282.

d. The Population Record

i. Essential variables:

- Housing unit (`serialno`)
- Person number (`sporder`)
- Person weight replicate (`pwgtp`)
- Person age (`agep`)
- Marital status (`mar`)
- Relationship (`rel`)
- Sex (`sex`)

ii. Stata code:

```
use "C:\psam_pusa.dta"
keep serialno sporder mar agep rel sex pwgtp*
count
sort serialno sporder
save "C:\psam_pusa_001.dta"
```

```
. use "R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_pus\psam_pusa.dta", clear

. keep serialno sporder mar agep rel sex pwgtp*

. count
1548768

. sort serialno sporder

. save "R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_pus\psam_pusa_001.dta"
file R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_pus\psam_pusa_001.dta saved
```

- Please note that the command `keep pwgtp*` retains <u>all</u> variables with that prefix: this includes not only the person weight replicate (`pwgtp`), but also all 80

population weights (`pwgtp1` – `pwgtp80`). These variables are necessary to generate weighted estimates.

- Take note of results for the `count` command
- Perform again using "C:\psam_pusb.dta"; results for `count` command should be equal to 1,451,889.

4. Append and Merge Data:
   a. Appending the Housing Records (**psam_husa_001.dta** & **psam_husb_001.dta**)
      i. Stata code:

```
use “C:\psam_husa_001.dta”, clear
merge serialno using “C:/psam_husb_001.dta”
tab _merge
drop _merge
sort serialno
save “C:\psam_husab_001.dta”
```

- Though technically an append, using the `merge` command will generate a useful variable (`_merge`) which indicates the number of cases read in from each file. The `tab _merge` command should correspond to results from the `count` commands in the previous steps.

```
. use "R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_hus\psam_husa_001.dta", clear

. merge serialno using "R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_hus\psam_husb_001.dta"

. tab _merge

    _merge |      Freq.     Percent        Cum.
-----------+-----------------------------------
         1 |    705,884       51.41       51.41
         2 |    667,282       48.59      100.00
-----------+-----------------------------------
     Total |  1,373,166      100.00

. drop _merge

. sort serialno

. save "R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_hus\psam_husab_001.dta"
file R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_hus\psam_husab_001.dta saved
```

b. Appending the Population Records (**psam_pusa_001.dta** & **psam_pusb_001.dta**)

i. Stata code:

```
use “C:\psam_pusa_001.dta”, clear
merge serialno sporder using “C:\psam_pusb_001.dta”
tab _merge
drop _merge
sort serialno sporder
save “C:\psam_pusab_001.dta”
```

```
. use "R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_pus\psam_pusa_001.dta", clear

. merge serialno sporder using "R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_pus\psam_pusb_001.dta"

. tab _merge

    _merge |      Freq.     Percent        Cum.
-----------+-----------------------------------
         1 |  1,548,768       51.61       51.61
         2 |  1,451,889       48.39      100.00
-----------+-----------------------------------
     Total |  3,000,657      100.00

. drop _merge

. sort serialno sporder

. save "R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_pus\psam_pusab_001.dta"
file R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_pus\psam_pusab_001.dta saved
```

c. Merging Population and Housing Records (one-to-many merge)

i. Stata code:

```
use “C:\psam_pusab_001.dta”, clear
merge serialno using “C:\psam_husab_001.dta”
tab _merge
sort serialno sporder
save “C:\psam_PusHus_001.dta”
```

- You must treat the Population Record as the master dataset (achieved through `use “C:\psam_pusab_001.dta”, clear`), otherwise you will lose information during the procedure

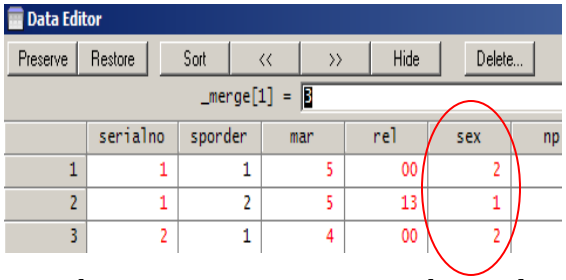
```
. use "R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_pus\psam_pusab_001.dta", clear

. merge serialno using "R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_hus\psam_husab_001.dta"
variable serialno does not uniquely identify observations in the master data

. tab _merge

    _merge |      Freq.     Percent        Cum.
-----------+-----------------------------------
         2 |    105,323        3.39        3.39
         3 |  3,000,657       96.61      100.00
-----------+-----------------------------------
     Total |  3,105,980      100.00

. sort serialno sporder

. save "R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\psam_PusHus_001.dta"
file R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\psam_PusHus_001.dta saved
```

ii. Once completed you might see this message…

```
. merge serialno using "R:\CFDR\CFDR\Ryan\Same-Sex Couples Work\PUMS_ACS2008_TEST\pc_hus\psam_husab_001.dta"
variable serialno does not uniquely identify observations in the master data
```

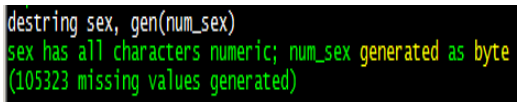
- Don’t be alarmed! This is normal, as `serialno` is certainly not unique in either the Housing or Population Records.
- The message is simply informative. If you have followed the procedure as outlined above the `merge` should have occurred without any problems.

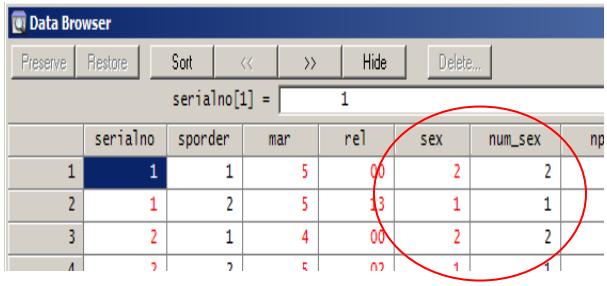
5. Convert to Numeric:
   a. Conversion from **string** to **numeric** format
      i. In most cases the variable values will automatically be **string** format (identified as **red** values in the data editor/browser) in your ACS download. These have to be converted to **numeric** format (identified as **black** values in the data editor/browser, or **blue** values if labels are attached) for the purposes of analysis; otherwise you will get a type mismatch.
         - This example uses the variable `sex`, however many variables in your downloaded data (including those used later in this example) are in **string** format and will have to be changed; use the following commands as needed.



      ii. When converting you may either replace **numeric** characters for **string** **OR** create a **numeric** analog to the given **string** variable
      iii. Stata code:
         - Create a **numeric** analog for a **string** variable

```
destring sex, gen (num_sex)
```

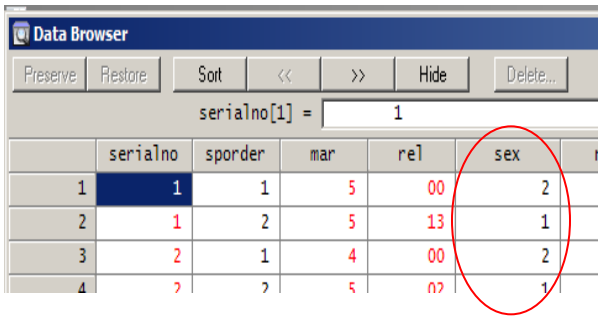




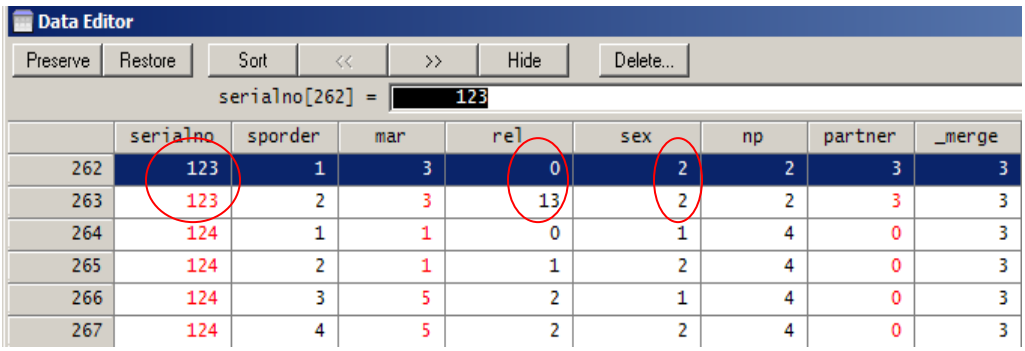
- **OR**, replace **string** for **numeric**

```
destring sex, replace
```

```
. destring sex, replace
sex has all characters numeric; replaced as byte
(105323 missing values generated)
```



6. Constructing Couple-Level Characteristics.
   a. This example is interested in identifying individuals in same-sex relationships, and examining differences between those in opposite-sex marriages and cohabitations.



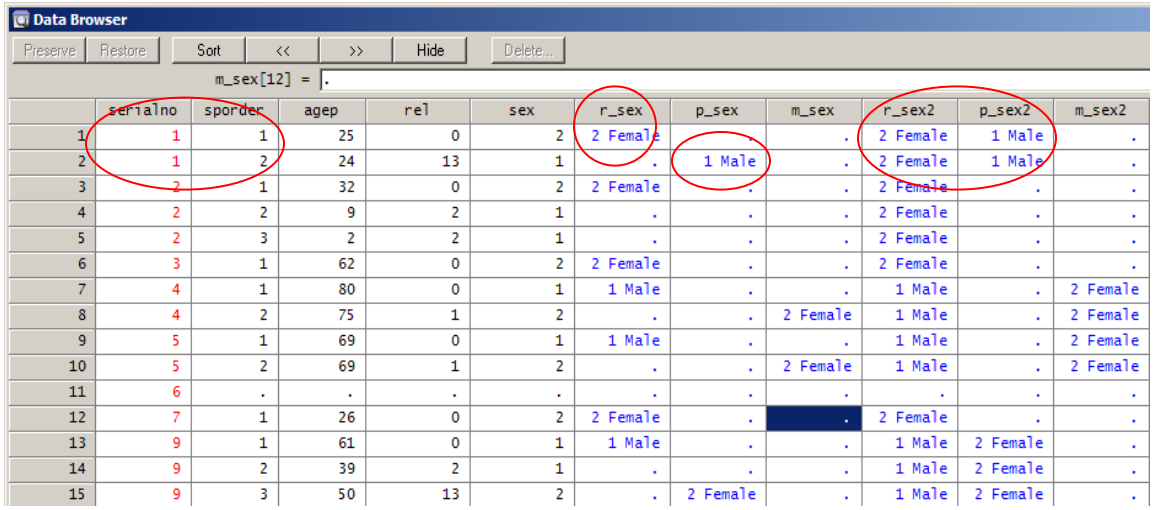
	serialno	sporder	mar	rel	sex	np	partner	_merge
262	123	1	3	0	2	2	3	3
263	123	2	3	13	2	2	3	3
264	124	1	1	0	1	4	0	3
265	124	2	1	1	2	4	0	3
266	124	3	5	2	1	4	0	3
267	124	4	5	2	2	4	0	3

   b. The variable `rel` indicates the relationship of each individual within a household with respect to the household head; individual households are identified by the variable `serialno`. Household 123 has two members; a household head (`sporder` = 1, `rel` = 0) and an unmarried partner (`sporder` = 2, `rel` = 13). Notice that the sex of the head and the unmarried partner is the same; this is a same-sex relationship. However, because this information is contained on **separate lines**, you cannot identify this characteristics without first manipulating the data.
   c. Identify the sex of the household head and the unmarried partner (or spouse), and recode as <u>three</u> new variables: `r_sex` (sex of household head), `p_sex` (sex of unmarried partner), and `m_sex` (sex of spouse).
      i. Stata code:

```
gen r_sex=sex if rel==00
gen p_sex=sex if rel==13
gen m_sex=sex if rel==01
```

   d. Create indicators of sex of the household head and spouse/unmarried partner which will be placed on the line of information for each individual within a specific household.
      i. Stata code:

```
by serialno: egen r_sex2=max(r_sex)
by serialno: egen p_sex2=max(p_sex)
by serialno: egen m_sex2=max(m_sex)
```


Data Browser

Preserve | Restore | Sort | << | >> | Hide | Delete...

m_sex[12] = .

	serialno	sporder	agep	rel	sex	r_sex	p_sex	m_sex	r_sex2	p_sex2	m_sex2
1	1	1	25	0	2	2 Female	.	.	2 Female	1 Male	.
2	1	2	24	13	1	.	1 Male	.	2 Female	1 Male	.
3	2	1	32	0	2	2 Female	.	.	2 Female	.	.
4	2	2	9	2	1	.	.	.	2 Female	.	.
5	2	3	2	2	1	.	.	.	2 Female	.	.
6	3	1	62	0	2	2 Female	.	.	2 Female	.	.
7	4	1	80	0	1	1 Male	.	.	1 Male	.	2 Female
8	4	2	75	1	2	.	.	2 Female	1 Male	.	2 Female
9	5	1	69	0	1	1 Male	.	.	1 Male	.	2 Female
10	5	2	69	1	2	.	.	2 Female	1 Male	.	2 Female
11	6	.	.	.	.	.	.	.	.	.	.
12	7	1	26	0	2	2 Female	.	.	2 Female	.	.
13	9	1	61	0	1	1 Male	.	.	1 Male	2 Female	.
14	9	2	39	2	1	.	.	.	1 Male	2 Female	.
15	9	3	50	13	2	.	2 Female	.	1 Male	2 Female	.

ii. The `egen` command created a new variable with values for each individual within a household (`by serialno`).

- For the first household (`serialno` = 1), the maximum value for `r_sex` (= 2) and the maximum value for `p_sex` (= 1) were placed in `r_sex2` and `p_sex2` respectively.
- Thus, the first individual in the first household (`serialno` = 1, `sporder` = 1), now has values of 2 and 1 for `r_sex2` and `p_sex2`, even though data is missing on `p_sex` for this individual.
- Please note that the above code does not include value labels, and your view of the data editor/browser (while still containing the correct values) will be different. Please refer to the attached .do file for value label codes.

e. Although couple-level characteristics are now included on individual lines of information, two more steps are required before the data is ready for analysis.

Data Browser

Preserve | Restore | Sort | << | >> | Hide | Delete...

m_sex[12] = .

	serialno	sporder	agep	rel	sex	r_sex	p_sex	m_sex	r_sex2	p_sex2	m_sex2
1	1	1	25	0	2	2 Female	.	.	2 Female	1 Male	.
2	1	2	24	13	1	.	1 Male	.	2 Female	1 Male	.
3	2	1	32	0	2	2 Female	.	.	2 Female	.	.
4	2	2	9	2	1	.	.	.	2 Female	.	.
5	2	3	2	2	1	.	.	.	2 Female	.	.
6	3	1	62	0	2	2 Female	.	.	2 Female	.	.
7	4	1	80	0	1	1 Male	.	.	1 Male	.	2 Female
8	4	2	75	1	2	.	.	2 Female	1 Male	.	2 Female
9	5	1	69	0	1	1 Male	.	.	1 Male	.	2 Female
10	5	2	69	1	2	.	.	2 Female	1 Male	.	2 Female
11	6	.	.	.	.	.	.	.	.	.	.
12	7	1	26	0	2	2 Female	.	.	2 Female	.	.
13	9	1	61	0	1	1 Male	.	.	1 Male	2 Female	.
14	9	2	39	2	1	.	.	.	1 Male	2 Female	.
15	9	3	50	13	2	.	2 Female	.	1 Male	2 Female	.

f. Take note of the ninth household (`serialno` = 9). It contains a biological son of the household head (`sporder` = 2, `rel` = 2). This individual is not in a romantic relationship with the head, and yet his values for `r_sex2` and `p_sex2` accurately reflect the sex of the head and the head's unmarried partner. If you were to use `r_sex2` and `p_sex2` to construct an indicator of relationship type, you would mistakenly identify this individual as engaged in opposite-sex cohabitation. Create a new variable to distinguish between those in a relationship with the head, and those who are just living in the household.

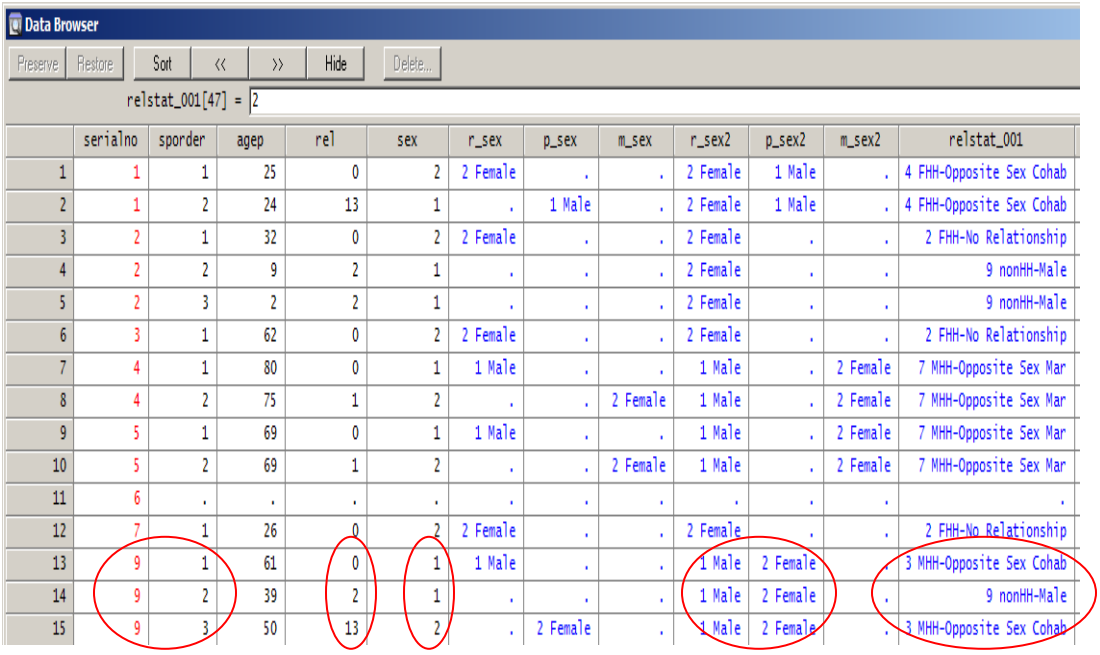
g. Use `r_sex2`, `p_sex2`, and `m_sex2` to create a complex indicator (`relstat_001`) of sex and relationship type where…

- 1 = no relationship, household head male
- 2 = no relationship, household head female
- 3 = opposite-sex cohabitation, household head male
- 4 = opposite-sex cohabitation, household head female
- 5 = same-sex cohabitation, household head male
- 6 = same-sex cohabitation, household head female
- 7 = opposite-sex marriage, household head male
- 8 = opposite-sex marriage, household head female
- 9 = no recorded romantic relationship, male householder
- 10 = no recorded romantic relationship, female householder
- 161 = male in institutional group quarters
- 162 = female in institutional group quarters
- 171 = male in non-institutional group quarters
- 172 = female in non-institutional group quarters

ii. Note that categories 1 – 8 are not gender-specific; these categories identify the type of romantic relationship by the sex of the household head. Categories 9 and 10 are gender-specific; these categories identify the sex of individuals in household who are not in a romantic relationship with the head. Categories 161 – 172 are also gender-specific; these categories identify the sex of individuals in institutional and non-institutional group quarters, with no set household head.

iii. Stata code:

```
gen relstat_001=.
replace relstat_001=1 if r_sex2==1 & p_sex2==. & m_sex2==.
replace relstat_001=2 if r_sex2==2 & p_sex2==. & m_sex2==.
replace relstat_001=3 if r_sex2==1 & p_sex2==2
replace relstat_001=4 if r_sex2==2 & p_sex2==1
replace relstat_001=5 if r_sex2==1 & p_sex2==1
replace relstat_001=6 if r_sex2==2 & p_sex2==2
replace relstat_001=7 if r_sex2==1 & m_sex2==2
replace relstat_001=8 if r_sex2==2 & m_sex2==1
replace relstat_001=9 if sex==1 & rel!=. & rel!=0 & rel!=1 & rel!=13 &
rel!=16 & rel!=17
replace relstat_001=10 if sex==2 & rel!=. & rel!=0 & rel!=1 & rel!=13 &
rel!=16 & rel!=17
replace relstat_001=161 if sex==1 & rel==16
replace relstat_001=162 if sex==2 & rel==16
replace relstat_001=171 if sex==1 & rel==17
replace relstat_001=172 if sex==2 & rel==17
```



	serialno	sporder	agep	rel	sex	r_sex	p_sex	m_sex	r_sex2	p_sex2	m_sex2	relstat_001
1	1	1	25	0	2	2 Female	.	.	2 Female	1 Male	.	4 FHH-Opposite Sex Cohab
2	1	2	24	13	1	.	1 Male	.	2 Female	1 Male	.	4 FHH-Opposite Sex Cohab
3	2	1	32	0	2	2 Female	.	.	2 Female	.	.	2 FHH-No Relationship
4	2	2	9	2	1	.	.	.	2 Female	.	.	9 nonHH-Male
5	2	3	2	2	1	.	.	.	2 Female	.	.	9 nonHH-Male
6	3	1	62	0	2	2 Female	.	.	2 Female	.	.	2 FHH-No Relationship
7	4	1	80	0	1	1 Male	.	.	1 Male	.	2 Female	7 MHH-Opposite Sex Mar
8	4	2	75	1	2	.	.	2 Female	1 Male	.	2 Female	7 MHH-Opposite Sex Mar
9	5	1	69	0	1	1 Male	.	.	1 Male	.	2 Female	7 MHH-Opposite Sex Mar
10	5	2	69	1	2	.	.	2 Female	1 Male	.	2 Female	7 MHH-Opposite Sex Mar
11	6	.	.	.	.	.	.	.	.	.	.	.
12	7	1	26	0	2	2 Female	.	.	2 Female	.	.	2 FHH-No Relationship
13	9	1	61	0	1	1 Male	.	.	1 Male	2 Female	.	3 MHH-Opposite Sex Cohab
14	9	2	39	2	1	.	.	.	1 Male	2 Female	.	9 nonHH-Male
15	9	3	50	13	2	.	2 Female	.	1 Male	2 Female	.	3 MHH-Opposite Sex Cohab

- Again, please note that the above commands will not produce value labels. Please refer to the attached .do file for value label codes.

h. Using information from `relstat_001`, now create a simple categorical indicator for use in analysis where…

- 0 = no romantic coresidential union
- 1 = opposite-sex marriage
- 2 = opposite-sex cohabitation
- 3 = same-sex cohabitation

ii. Stata code:

```
gen relstat_002=.
replace relstat_002=0 if (relstat_001==1|relstat_001==2|relstat_001==9|
relstat_001==10|relstat_001==161|relstat_001==162|relstat_001==171|
relstat_001==172)
replace relstat_002=1 if (relstat_001==7|relstat_001==8)
replace relstat_002=2 if (relstat_001==3|relstat_001==4)
replace relstat_002=3 if (relstat_001==5|relstat_001==6)
```

- Now only those actually in some coresidential relationship (at least those in one with the household head) are recorded as such

Data Browser

Preserve | Restore | Sort | << | >> | Hide | Delete...

relstat_001[47] = 2

	serialno	sporder	agep	rel	sex	r_sex	p_sex	m_sex	r_sex2	p_sex2	m_sex2	relstat_001	relstat_002
1	1	1	25	0	2	2 Female	.	.	2 Female	1 Male	.	4 FHH-Opposite Sex Cohab	2
2	1	2	24	13	1	.	1 Male	.	2 Female	1 Male	.	4 FHH-Opposite Sex Cohab	2
3	2	1	32	0	2	2 Female	.	.	2 Female	.	.	2 FHH-No Relationship	0
4	2	2	9	2	1	.	.	.	2 Female	.	.	9 nonHH-Male	0
5	2	3	2	2	1	.	.	.	2 Female	.	.	9 nonHH-Male	0
6	3	1	62	0	2	2 Female	.	.	2 Female	.	.	2 FHH-No Relationship	0
7	4	1	80	0	1	1 Male	.	.	1 Male	.	2 Female	7 MHH-Opposite Sex Mar	1
8	4	2	75	1	2	.	.	2 Female	1 Male	.	2 Female	7 MHH-Opposite Sex Mar	1
9	5	1	69	0	1	1 Male	.	.	1 Male	.	2 Female	7 MHH-Opposite Sex Mar	1
10	5	2	69	1	2	.	.	2 Female	1 Male	.	2 Female	7 MHH-Opposite Sex Mar	1
11	6	.	.	.	.	.	.	.	.	.	.	.	.
12	7	1	26	0	2	2 Female	.	.	2 Female	.	.	2 FHH-No Relationship	0
13	9	1	61	0	1	1 Male	.	.	1 Male	2 Female	.	3 MHH-Opposite Sex Cohab	2
14	9	2	39	2	1	.	.	.	1 Male	2 Female	.	9 nonHH-Male	0
15	9	3	50	13	2	.	2 Female	.	1 Male	2 Female	.	3 MHH-Opposite Sex Cohab	2
908	416	1	55	0	2	2 Female	.	.	2 Female	2 Female	.	6 FHH-Same Sex Cohab	3
909	416	2	54	13	2	.	2 Female	.	2 Female	2 Female	.	6 FHH-Same Sex Cohab	3

- For example, the ninth household (`serialno` = 9) has three members. The head (`sporder` = 1) and the unmarried partner (`sporder` = 3) receive values of 2 for `relstat_002`, indicating that they are both in opposite-sex cohabiting relationships. The other member of the household (`sporder` = 2) is not in a recorded relationship and therefore receives a 0 for `relstat_002`.
- The data is now ready for further analysis.

7. Applying Weights in ACS
   a. Applying weights in ACS is essential for generating accurate population estimates. Since you constructed the indicator for union type using the Population Record, you will want to use the population weight (`pwgtp`) and the population weight replicates (`pwgtp1` – `pwgtp80`):
   b. Set your data for survey weights
      i. Stata code:

```
svyset [iw=pwgtp], jkrweight(pwgtp1-pwgtp80, multiplier (.05))
vce(jackknife) mse
```

      - Utilizing `svyset` will not fundamentally change your data in any way
      - To perform analyses using the weights you will have to recall the `svy` prefix for any proceeding commands

```
. svyset [iw=pwgtp], jkrweight(pwgtp1-pwgtp80, multiplier (.05)) vce(jackknife) mse

      iweight: pwgtp
          VCE: jackknife
          MSE: on
    jkrweight: pwgtp1 pwgtp2 pwgtp3 pwgtp4 pwgtp5 pwgtp6 pwgtp7 pwgtp8 pwgtp9 pwgtp10 pwgtp11 pwgtp12 pwgtp13 pwgtp14 pwgtp15
               pwgtp16 pwgtp17 pwgtp18 pwgtp19 pwgtp20 pwgtp21 pwgtp22 pwgtp23 pwgtp24 pwgtp25 pwgtp26 pwgtp27 pwgtp28 pwgtp29
               pwgtp30 pwgtp31 pwgtp32 pwgtp33 pwgtp34 pwgtp35 pwgtp36 pwgtp37 pwgtp38 pwgtp39 pwgtp40 pwgtp41 pwgtp42 pwgtp43
               pwgtp44 pwgtp45 pwgtp46 pwgtp47 pwgtp48 pwgtp49 pwgtp50 pwgtp51 pwgtp52 pwgtp53 pwgtp54 pwgtp55 pwgtp56 pwgtp57
               pwgtp58 pwgtp59 pwgtp60 pwgtp61 pwgtp62 pwgtp63 pwgtp64 pwgtp65 pwgtp66 pwgtp67 pwgtp68 pwgtp69 pwgtp70 pwgtp71
               pwgtp72 pwgtp73 pwgtp74 pwgtp75 pwgtp76 pwgtp77 pwgtp78 pwgtp79 pwgtp80
  Single unit: missing
     Strata 1: <one>
         SU 1: <observations>
        FPC 1: <zero>
```

   c. Try a simple crosstab (`agep` by `relstat_002`) using the weights
      i. Stata code:

```
svy: tab agep relstat_002, count cellwidth(20) format(%15.2g)
```

```
. svy: tab agep relstat_002, count cellwidth(20) format(%15.2g)
no room to add more variables because of width
    An attempt was made to add a variable that would have increased the memory required to store an observation beyond what is
    currently possible.  You have the following alternatives:

    1.  Store existing variables more efficiently; see help compress.

    2.  Drop some variables or observations; see help drop.  (Think of Stata's data area as the area of a rectangle; Stata can trade
        off width and length.)

    3.  Increase the amount of memory allocated to the data area using the set memory command; see help memory.
r(902);
```

ii. What does the error mean? Applying weights means that Stata must perform an estimation for every category of `agep` (1-98) by every category of `relstat_002` (0-3). Stata does not have enough memory (RAM) to perform this command. What should you do?

d. Say you are only interested in a certain cluster of ages (24 to 33 years in this example). The variable should be modified to account for Stata's memory limitations.

e. Create a new variable which collapses age

i. Stata code:

```
gen agep_002=agep
replace agep_002=0 if agep<24 & agep!=.
replace agep_002=1 if agep>33 & agep!=.
```

```
. gen agep_002=agep
(105323 missing values generated)

. replace agep_002=0 if agep<24 & agep!=.
(875739 real changes made)

. replace agep_002=1 if agep>33 & agep!=.
(1748860 real changes made)

. tab agep_002
```

agep_002	Freq.	Percent	Cum.
0	910,861	30.36	30.36
1	1,748,860	58.28	88.64
24	32,907	1.10	89.73
25	34,307	1.14	90.88
26	34,392	1.15	92.02
27	34,482	1.15	93.17
28	34,909	1.16	94.34
29	33,651	1.12	95.46
30	34,918	1.16	96.62
31	33,590	1.12	97.74
32	33,681	1.12	98.86
33	34,099	1.14	100.00
Total	3,000,657	100.00	

f. Try running the `svy` command again, but this time using your variable of age with collapsed categories

   i. **CAUTION**: The following command takes a considerable amount of time to run (approximately 30 minutes), make sure to plan accordingly.

   ii. Stata code:

```
svy: tab agep_002 relstat_002, count cellwidth(20) format(%15.2g)
```

```
. svy: tab agep_002 relstat_002, count cellwidth(20) format(%15.2g)
(running tabulate on estimation sample)

Number of strata   =         1             Number of obs      =    3000657
                                           Population size    =  304059728
                                           Replications       =         80
                                           Design df          =          4
```

agep_002	relstat_002 0	1	2	3	Total
0	97166191	1442347	1474260	46113	100128911
1	62246851	94583099	5777156	917341	163524447
24	2748542	718372	433548	12665	3913127
25	2842833	1028283	509997	11120	4392233
26	2536618	1210423	497431	19026	4263498
27	2316556	1373559	469194	14004	4173313
28	2136894	1587926	442826	16970	4184616
29	1835684	1663016	400012	21839	3920551
30	1886901	1844826	379253	17006	4127986
31	1578191	1860152	349739	21451	3809533
32	1542907	1952438	313233	18540	3827118
33	1429623	2050358	295729	18685	3794395
Total	180267791	111314799	11342378	1134760	304059728

```
  Key:  weighted counts
Note: variance estimate degrees of freedom = 4are less than nominal table degrees of freedom = 33

  Pearson:
    Uncorrected   chi2(33)         =  9.82e+05
    Design-based  F(22.91, 91.66)  =  1.70e+04     P = 0.0000
```

- Please note that the command `count cellwidth(20) format (%15.2g)` which follows the `tab` command is atypical; this command is not necessary for un-weighted data. However, the `count` option is required in order to generate the weighted number of individuals in each cell. Options `cellwidth(20)` and `format(%15.2g)` ensure that the results will be displayed properly.

8. Using the accompanying .do file:
   a. The attached .do file contains the appropriate commands to perform the procedures and analyses outlined above once you have obtained your ACS data.
   b. READ THROUGH THE .do PRIOR TO RUNNING!!!
   c. You will have to change the path names, and the list of variables you want to keep or drop will vary from those used in this example.
   d. The preceding outline gives you several options for structuring your data; make sure that the .do reflects your desired structure.
   e. The attached .do also contains commands for variable and value labels; these were not illustrated above.

9. Additional CFDR Guides for Analyzing Census Data:
   a. To append ACS files using SAS, please see http://www.bgsu.edu/downloads/cas/file97318.pdf.
   b. To append ACS files using Stata, please see http://www.bgsu.edu/downloads/cas/file97319.pdf.
   c. To attach spouse information in the Current Population Survey (CPS) using SAS, please see http://www.bgsu.edu/downloads/cas/file37390.pdf.
   d. To create cohabitation files in the Public Use Microdata Samples (PUMS) using SAS, please see http://www.bgsu.edu/downloads/cas/file37389.pdf.
   e. For additional guidance on using weights in ACS (using both SAS and Stata), please see http://www.bgsu.edu/downloads/cas/file75747.pdf.